

**CZ3006 Net-Centric Computing**

**Lab Assignment II:**

**Web applications using JavaScript and PHP**

Lab Group: TS2

Index Number: TS2U6

Name: Elroy Liang Bo Jun (U1720486H)

**Description of HTML**

1. **Web UI Design of the fruit order form page**

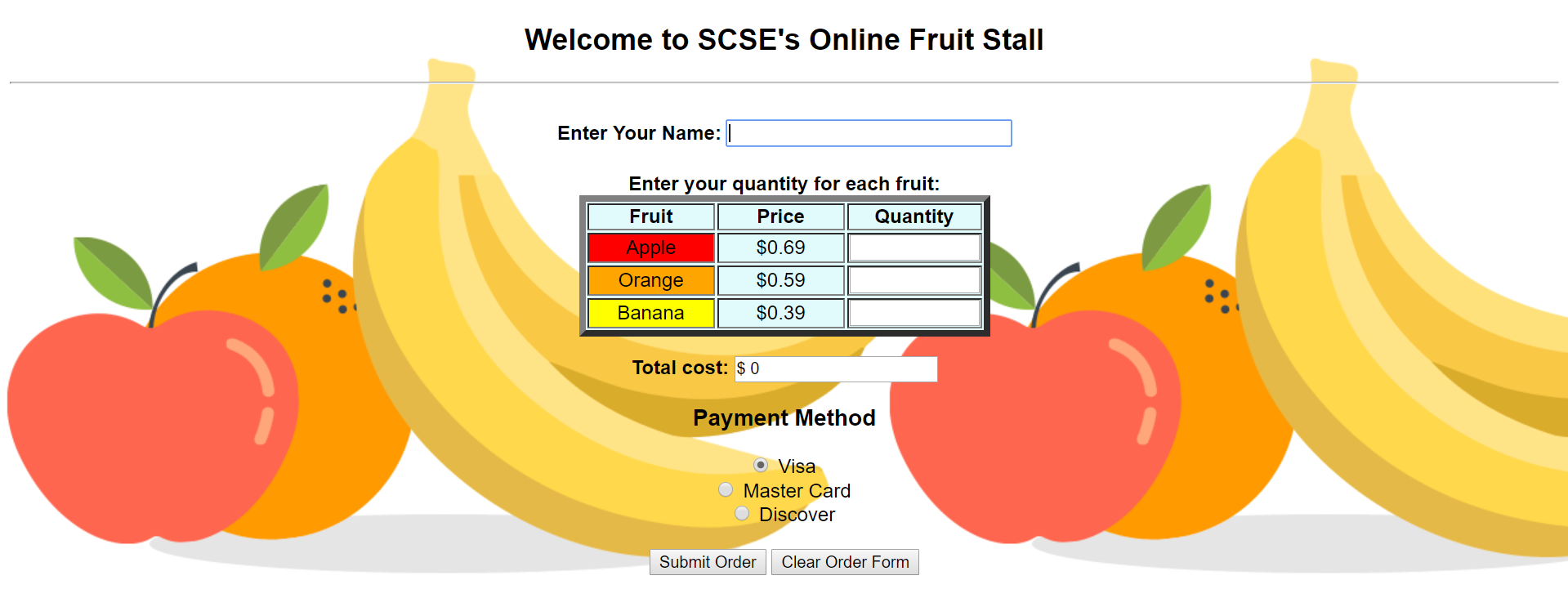


Figure 1: Order form page

The design of the fruit order form is designed to be simple and intuitive for users. In sequence from top to bottom, users will first enter their names, then enter the quantity of each fruit they wish to order, and lastly choose their payment method and submit form.

1. **Validate quantity of fruits textboxes**

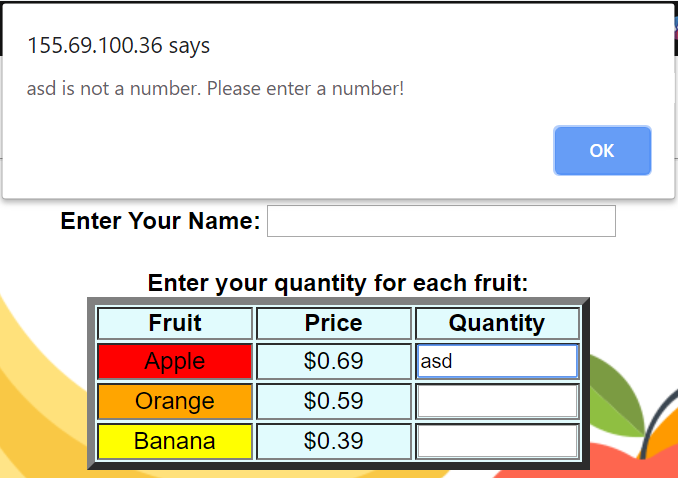


Figure 2.1: Javascript alert - input is not a number

Since quantity of fruits must be a number, we can allow user to input digits. When user inputs a non-positive number in the “Quantity” field textboxes, upon changing focus from the field, an alert will be prompted to request user to re-enter the value, as shown in Fig 2.1.

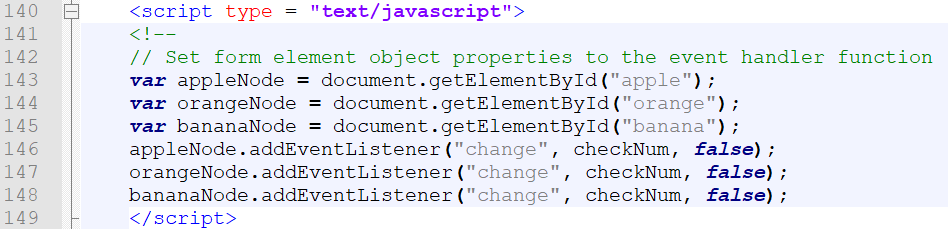


Figure 2.2: Set form element object properties to event handler function

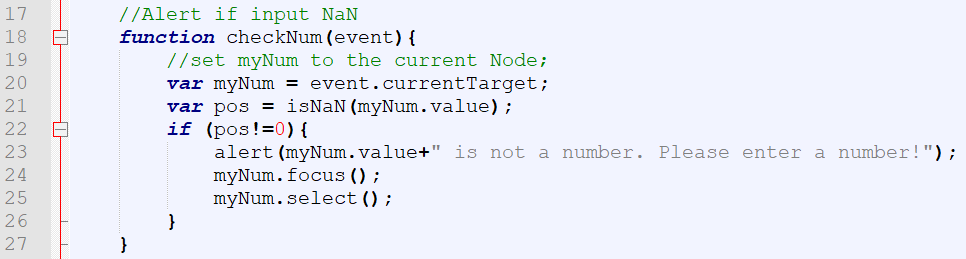


Figure 2.3: Implementation of ‘checkNum(event)’ in JavaScript

In figure 2.2, the form elements are set to the event handler function. Whenever a ‘change’ in focus from the element occuers, it will trigger the *checkNum(event)* function to run. The *checkNum(event)* function is shown in figure 2.3, in which it checks that the value passed is a number. In the case of figure 2.1, since the input is not a number, an alert will be prompted to ask user to re-input a valid number by re-focusing and selecting the invalid field.

1. **Calculation of total cost**

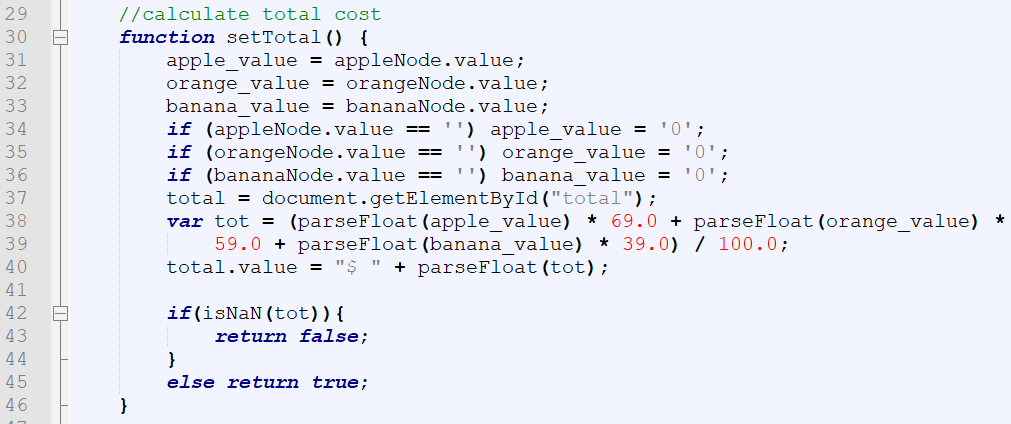


Figure 3.1: Implementation of ‘setTotal()’ in JavaScript

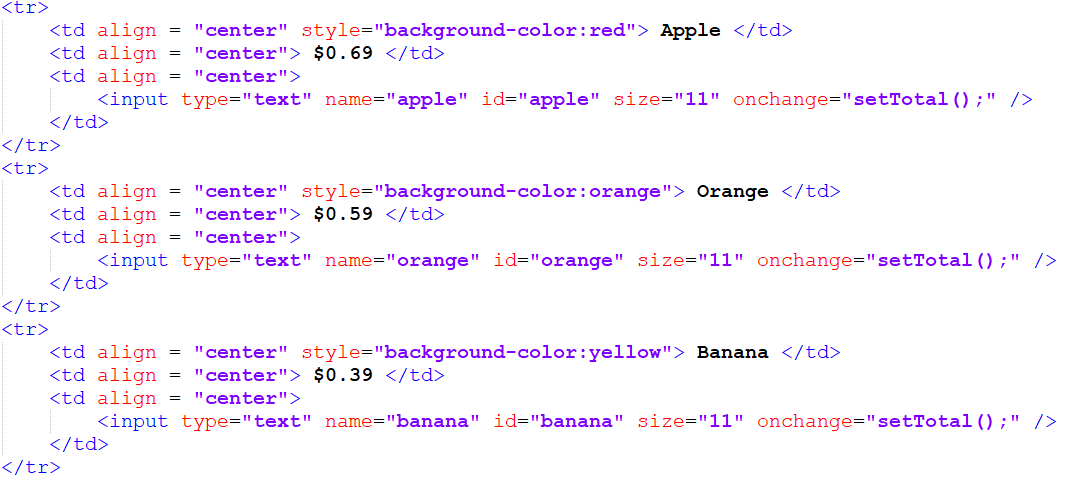


Figure 3.2: Implementation of ‘onchange=”setTotal(“)’ in html

The total cost of all the fruits are calculated based on the function *setTotal()*, as shown in figure 3.1*.* The function takes in the values of the orange, apple and banana objects, multiply each value with its corresponding price and sums up the results to get the total cost of all the fruits. To dynamically update the total cost of the fruits, the *onchange = ”setTotal()”* property is added in each of the input elements of each fruit as shown in figure 3.2. Whenever there is a change in input value of the quantity of each fruit, the new value of the total cost will be calculated.

1. **Validation on submission of form**

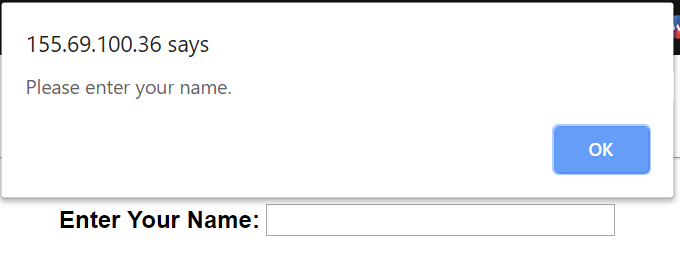


Figure 4.1: Javascript alert – name field is empty



Figure 4.2: Implementation of “onsubmit = return validate()” in html

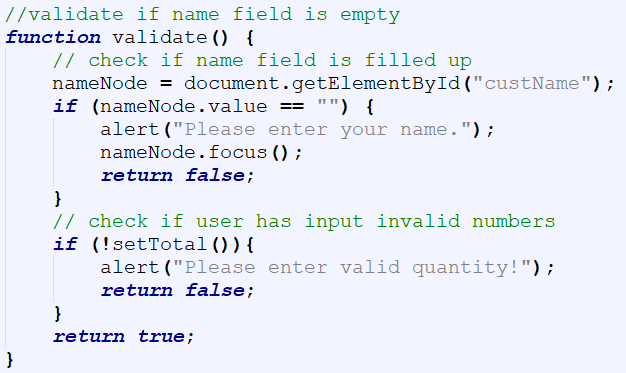


Figure 4.3: Implementation of validate() function in JavaScript

An alert will be prompted if user tries to submit the form without entering their name or with a “NaN” total cost value. This is implemented by setting *onsubmit=”return validate()”* in the form element shown in figure 4.2. The function *validate()* is shown in figure 4.3 in JavaScript, whereby it checks if the name value is empty and also checks if the total cost is “NaN”. Whenever one of them is true, an alert will be prompted, and user will be unable to submit the form.

**Description of PHP**

1. **Read and writing new values into the file**

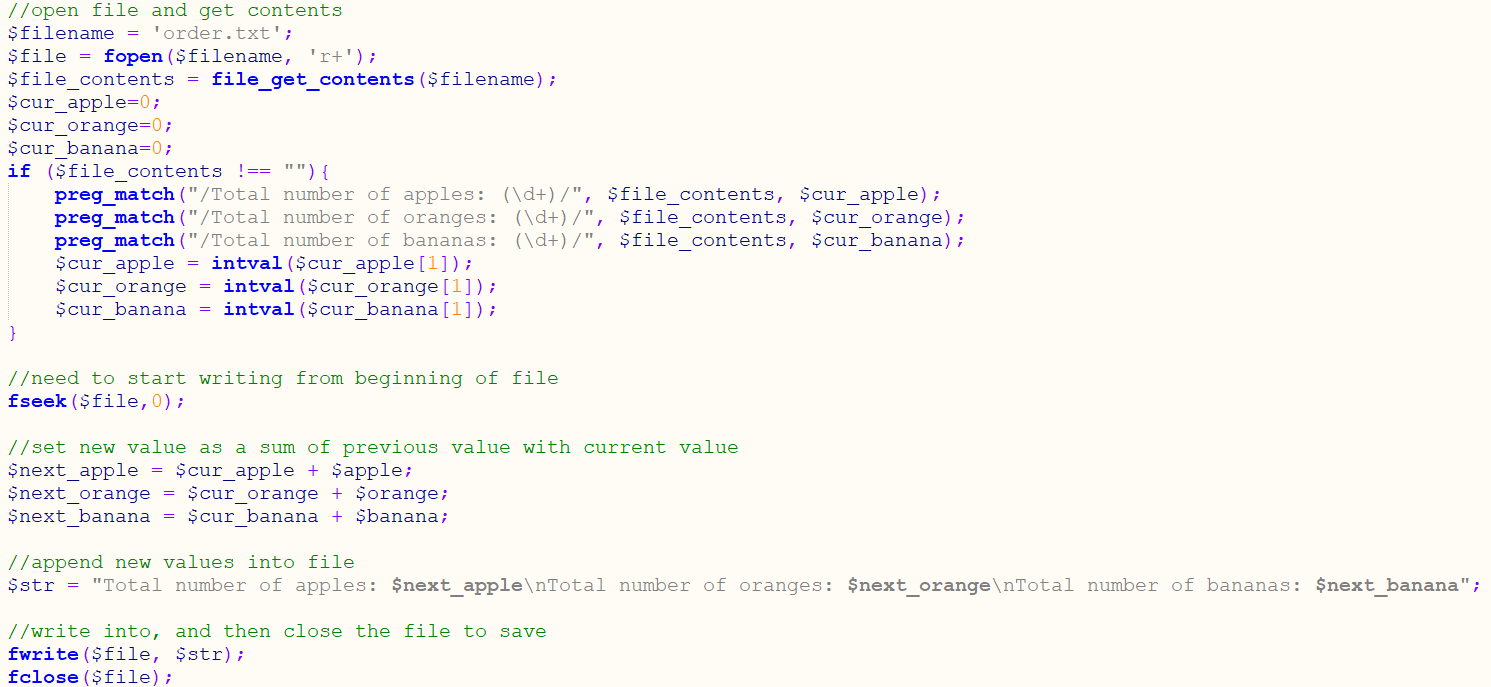


Figure 5.1: PHP implementation of modifying file

As shown clearly in figure 5.1, the file ‘order.txt’ is opened and its contents are fetched and assign to the respective pointers cur\_apple, cur\_orange and cur\_banana. These values are summed with the new values and the sum is stored in the respective pointers next\_apple next\_orange and next\_banana. These updated sum of each total number of fruits will be appended into the file by overwriting all the original values in the file.

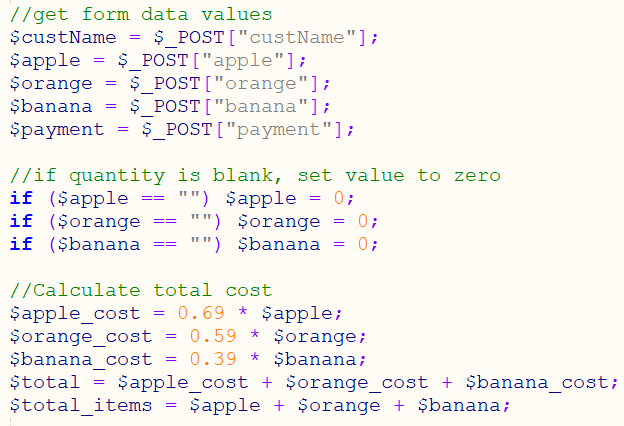
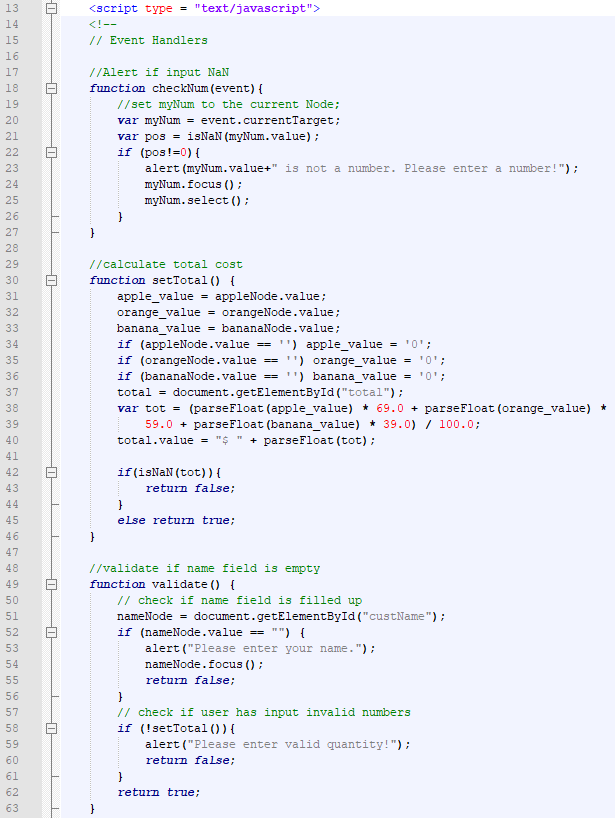


Figure 5.1: PHP implementation of calculating total cost

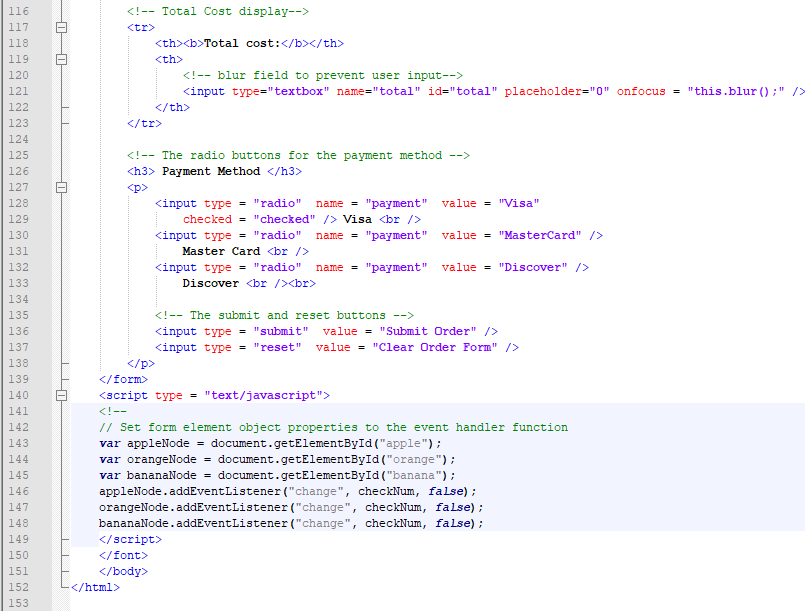
The figure above shows the implementation of getting the values from the html form and using them to calculate the total cost of all fruits. If any of the values are empty, a value of ‘0’ is set.

**Listing all source files**

1. **fruitstall.html**







1. **fruitstall.php**

